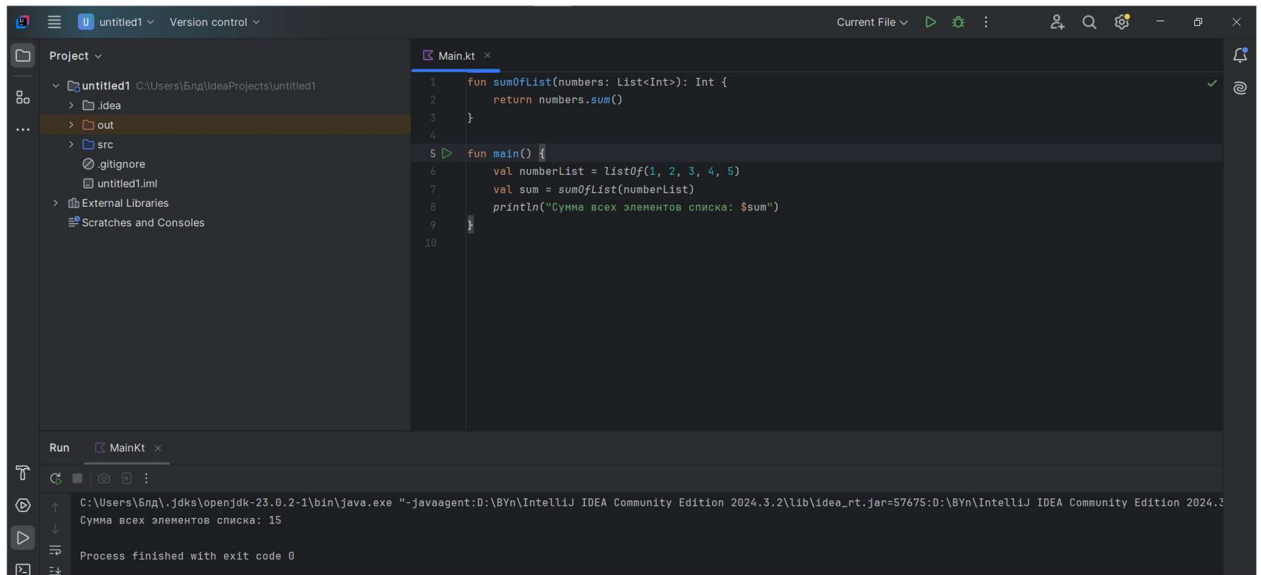


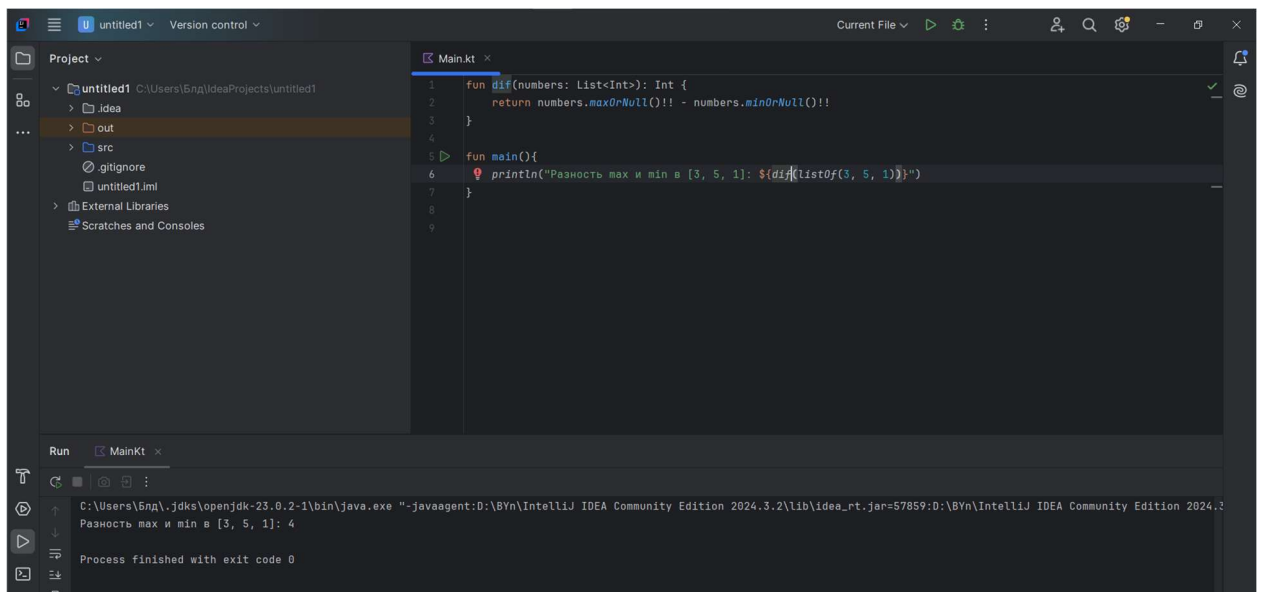
Лабораторная №8

1. Напишите функцию, которая принимает список на вход, и возвращает сумму всех элементов этого списка.



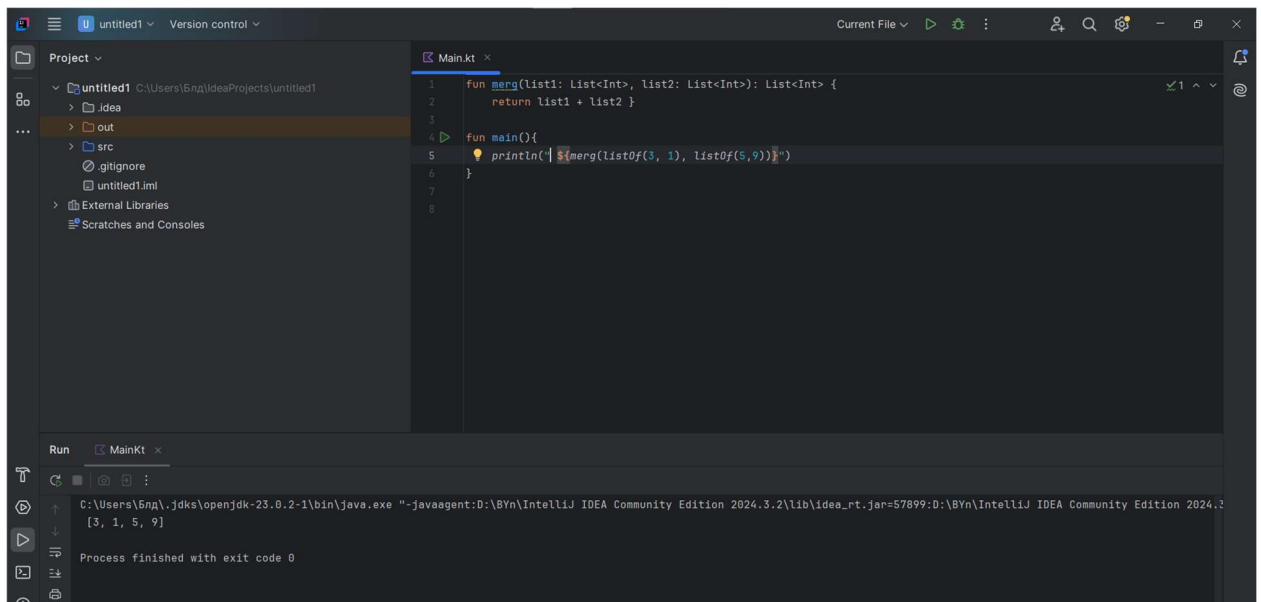
```
fun sumOfList(numbers: List<Int>): Int {
    return numbers.sum()
}
```

2. Напишите функцию, которая получает на вход список целых чисел и возвращает разность самого большого и самого маленького из них.



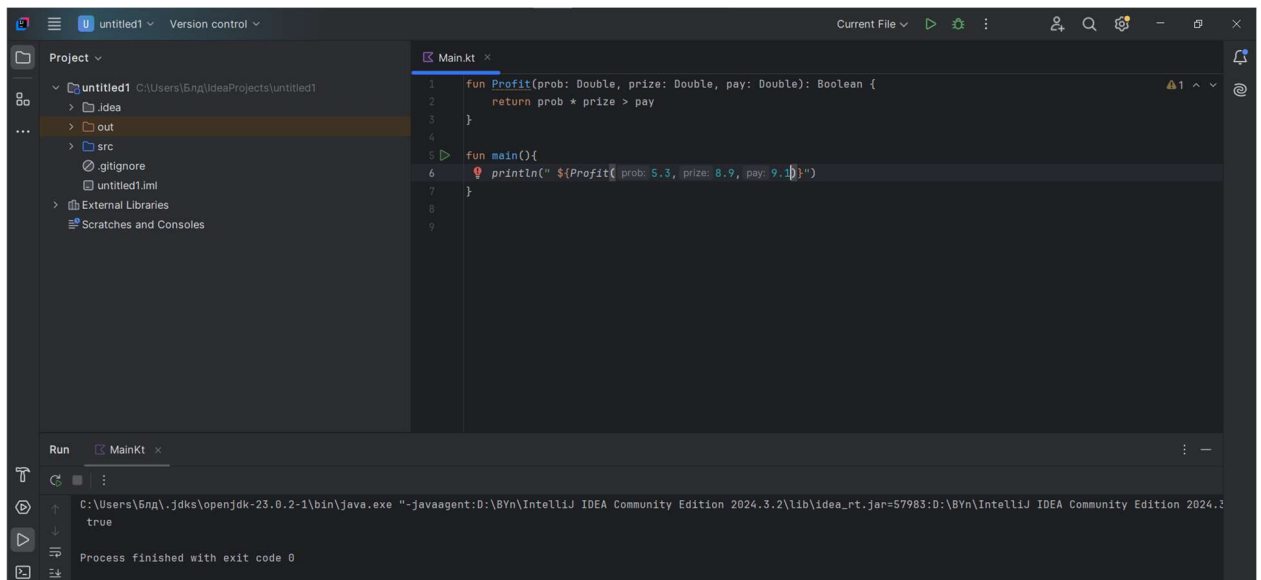
```
fun dif(numbers: List<Int>): Int {
    return numbers.maxOrNull()!! - numbers.minOrNull()!!
}
```

3. Создайте функцию для объединения двух списков целых чисел.



```
fun merg(list1: List<Int>, list2: List<Int>): List<Int> {  
    return list1 + list2  
}
```

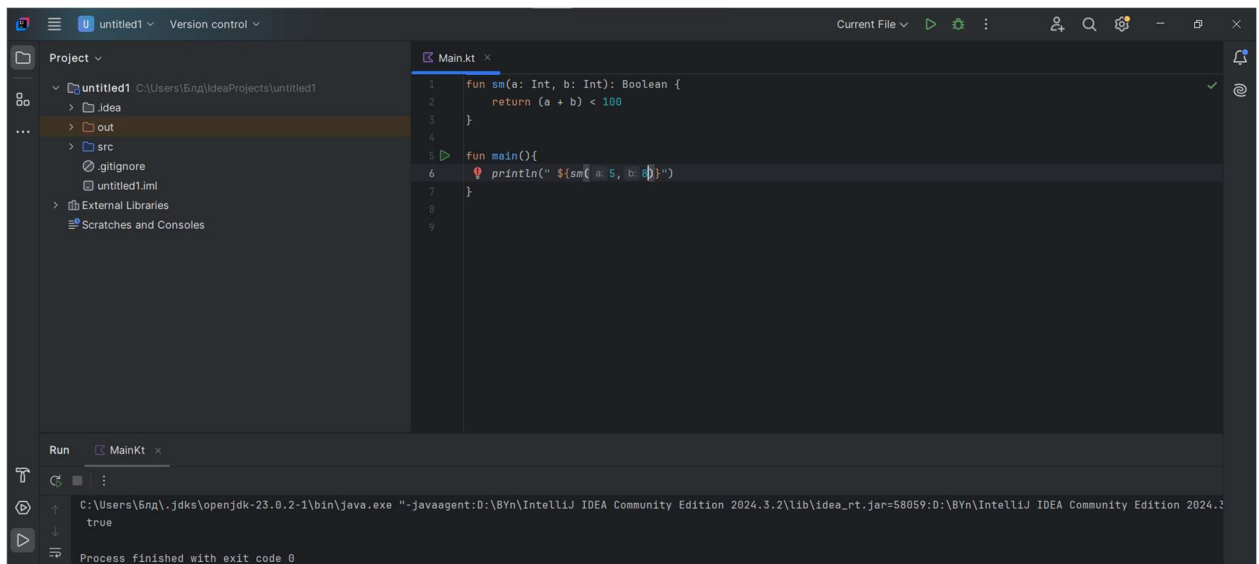
4. Создайте функцию, которая принимает три аргумента prob, prize, pay и возвращает True, если $\text{prob} * \text{prize} > \text{pay}$, в противном случае возвращает False.



```
fun Profit(prob: Double, prize: Double, pay: Double): Boolean {  
    return prob * prize > pay  
}
```

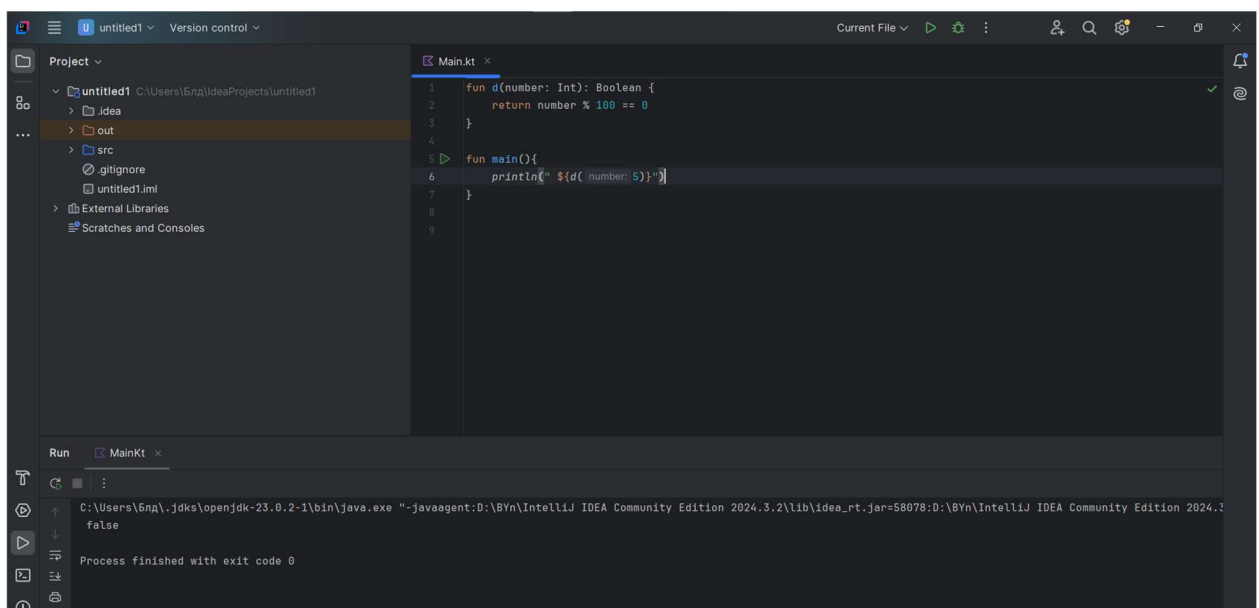
5.(тоже что и 4)

6. Функция получает на вход два числа. Она должна вернуть True, если сумма этих чисел меньше 100 и False в противном случае.



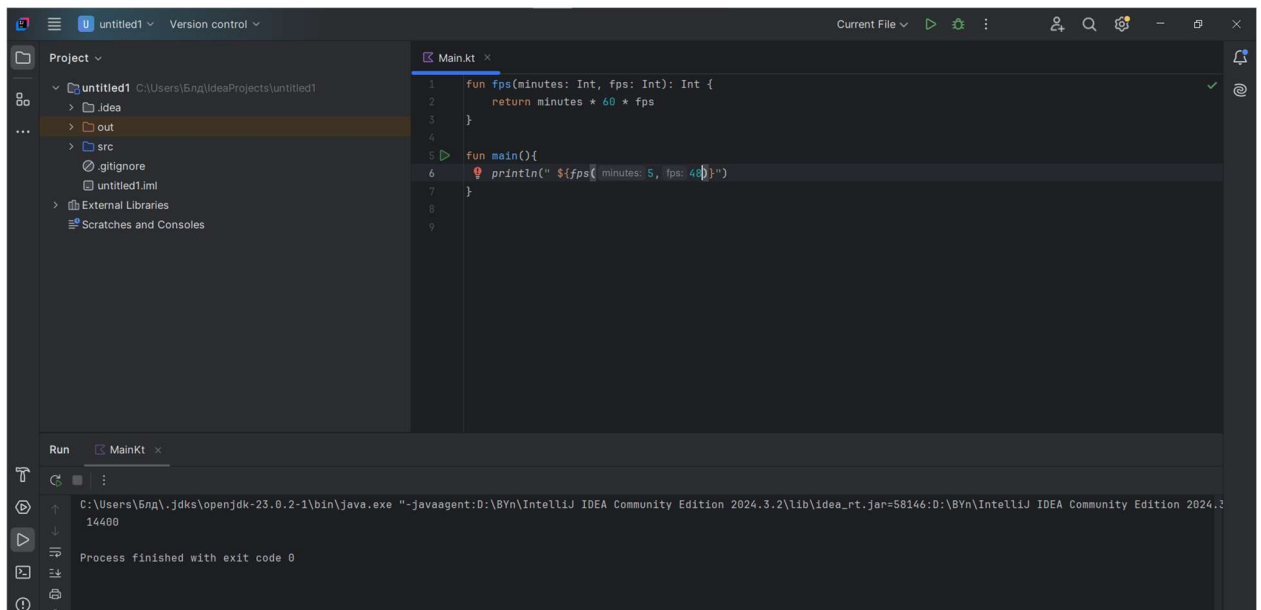
```
fun sm(a: Int, b: Int): Boolean {
    return (a + b) < 100
}
```

7. Напишите функцию, которая принимает целое число и возвращает True, если оно делится на 100. В противном случае функция должна вернуть False.



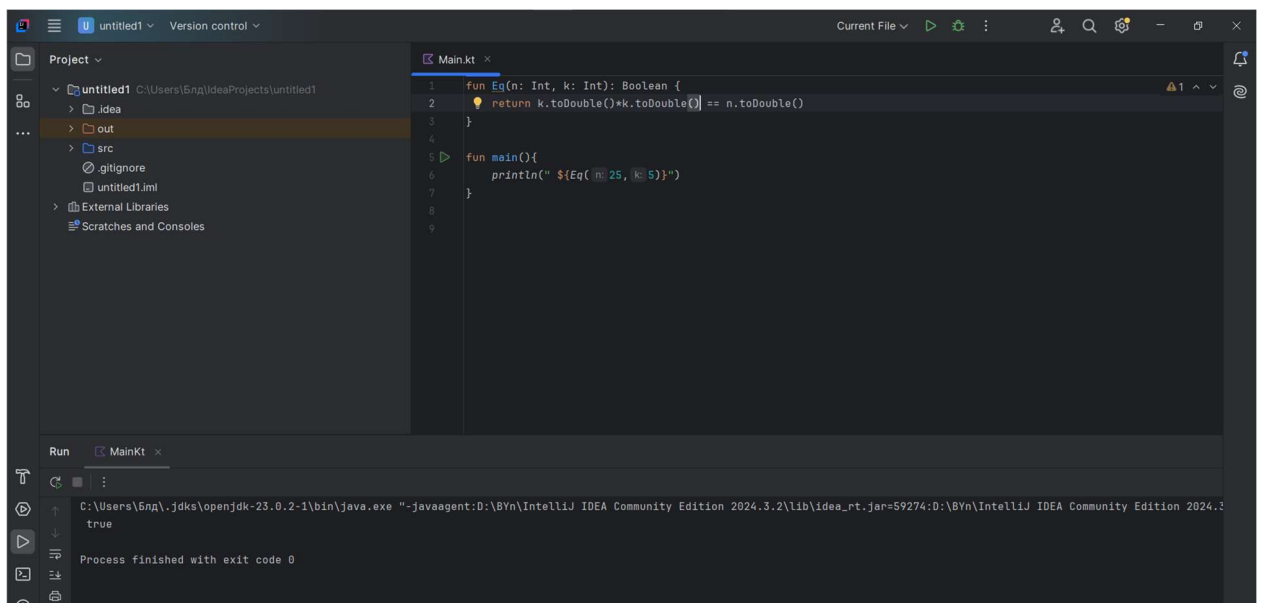
```
fun d(number: Int): Boolean {
    return number % 100 == 0
}
```

8. Напишите функцию, которая принимает количество минут и частоту кадров (FPS) и возвращает, сколько за это время кадров показывает компьютер при этом FPS.



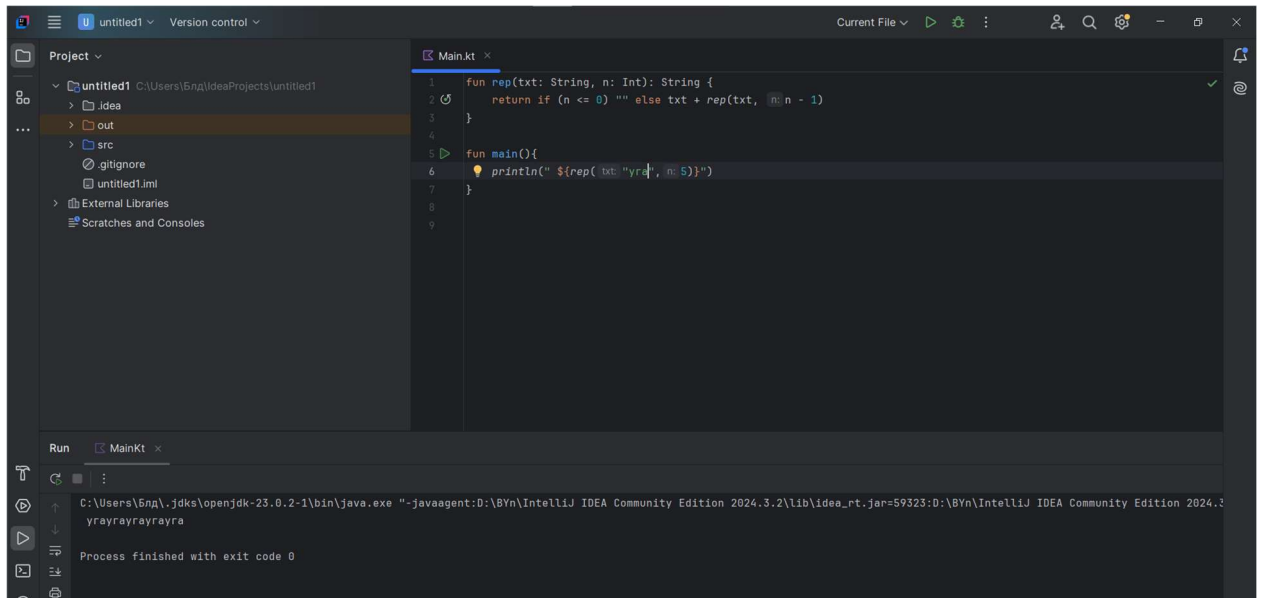
```
fun fps(minutes: Int, fps: Int): Int {
    return minutes * 60 * fps
}
```

9. Напишите функцию, которая возвращает True, если $k^k == n$ для входных данных (n, k), и возвращает False в противном случае.



```
fun Eq(n: Int, k: Int): Boolean {
    return k.toDouble()*k.toDouble() == n.toDouble()
}
```

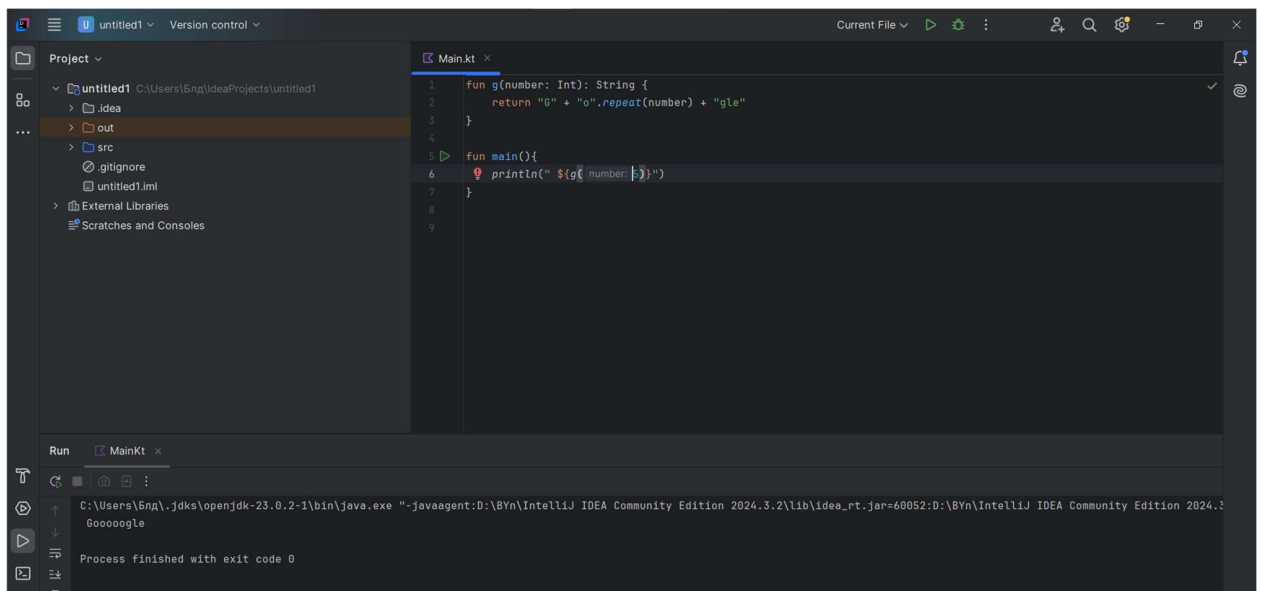
10. Создайте рекурсивную функцию, которая принимает два параметра и повторяет строку n количество раз. Первый параметр txt - это строка, которую нужно повторить, а второй параметр - количество повторений строки.



```
fun rep(txt: String, n: Int): String {
    return if (n <= 0) "" else txt + rep(txt, n - 1)
}
```

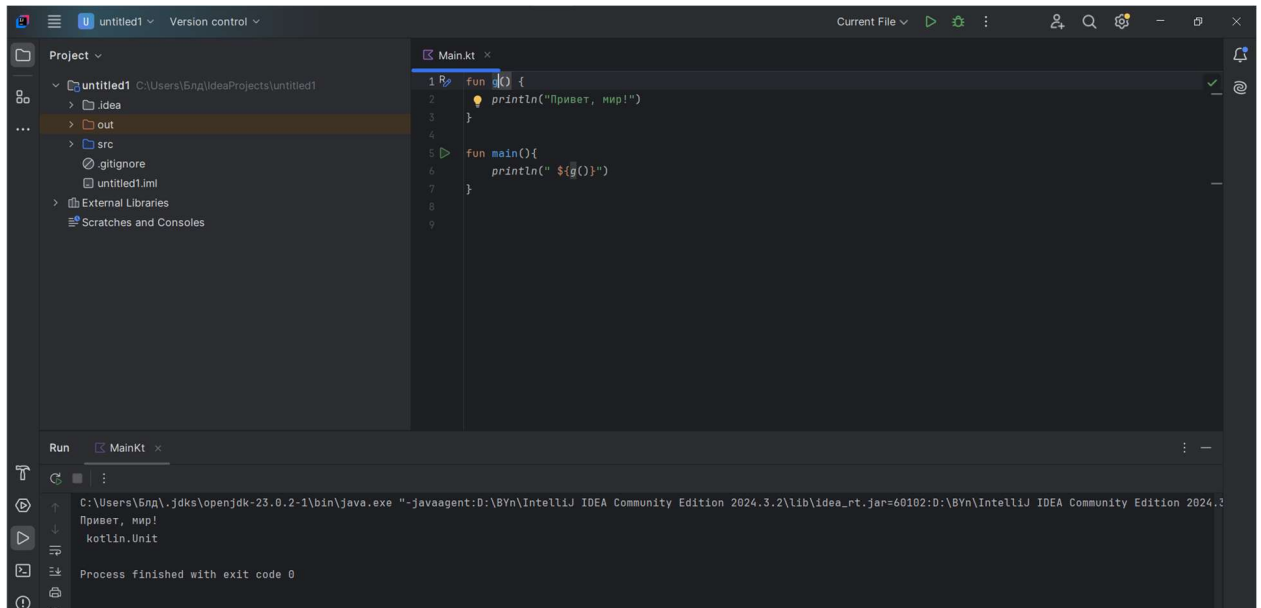
11. Создайте функцию, которая принимает уравнение (например, "1+1") и возвращает ответ.

12. Напишите функцию, которая принимает число number, и возвращает слово Google с количеством букв o, равным number



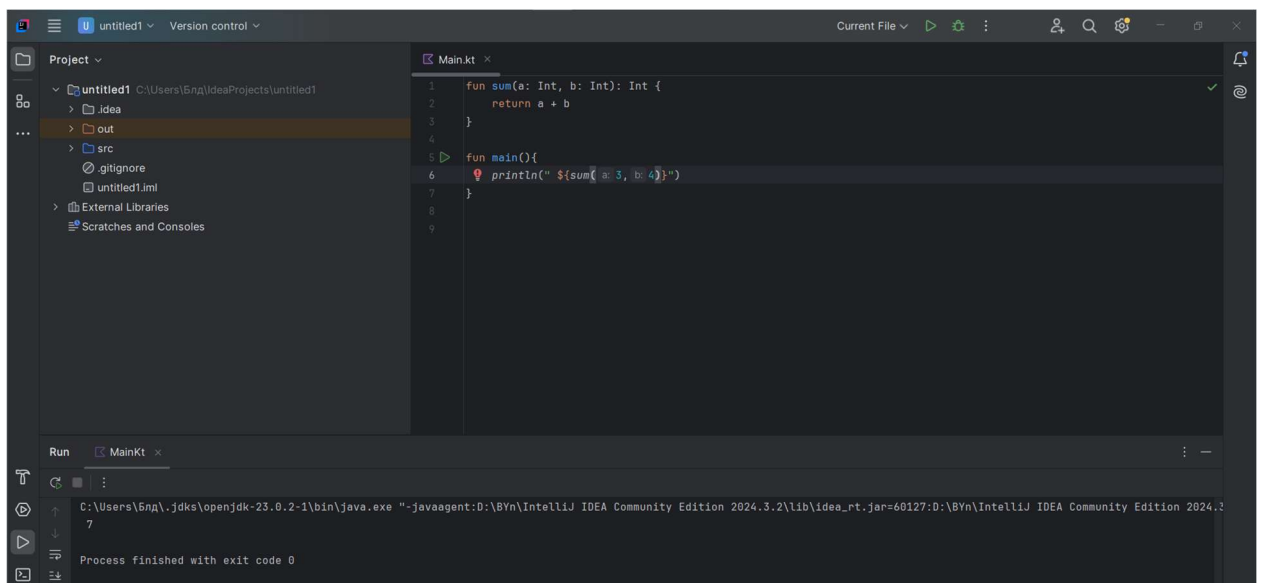
```
fun g(number: Int): String {
    return "G" + "o".repeat(number) + "gle"
}
```

13. Приветствие: Напишите функцию, которая выводит "Привет, мир!" на экран.



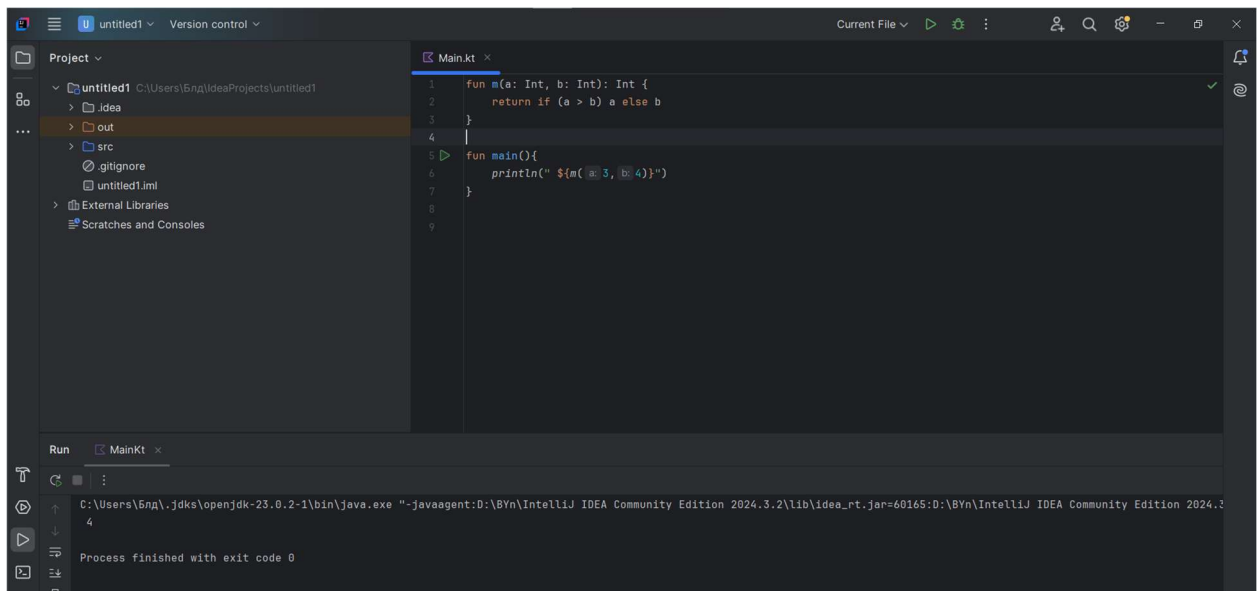
```
fun g() {
    println("Привет, мир!")
}
```

14. .Сумма двух чисел: Создайте функцию, которая принимает два числа и возвращает их сумму



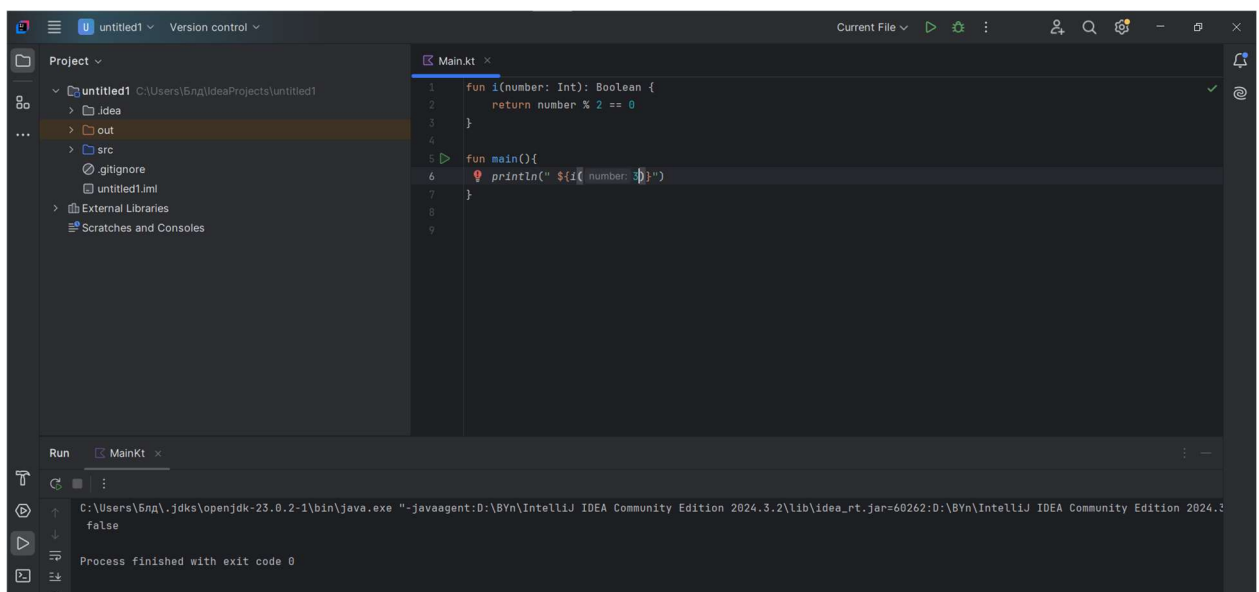
```
fun sum(a: Int, b: Int): Int {
    return a + b
}
```

15. Сравнение чисел: Напишите функцию, которая принимает два числа и возвращает большее из них.



```
fun m(a: Int, b: Int): Int {
    return if (a > b) a else b
}
```

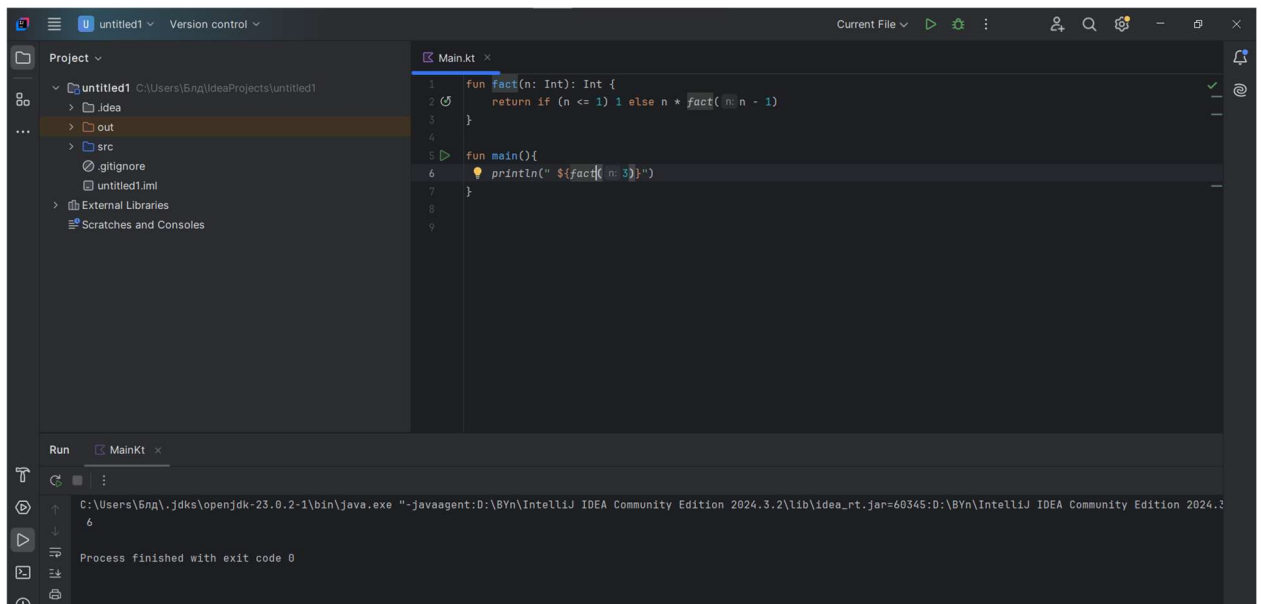
16. Определение четности: Создайте функцию, которая принимает одно число и возвращает true, если оно четное, и false, если нечетное.



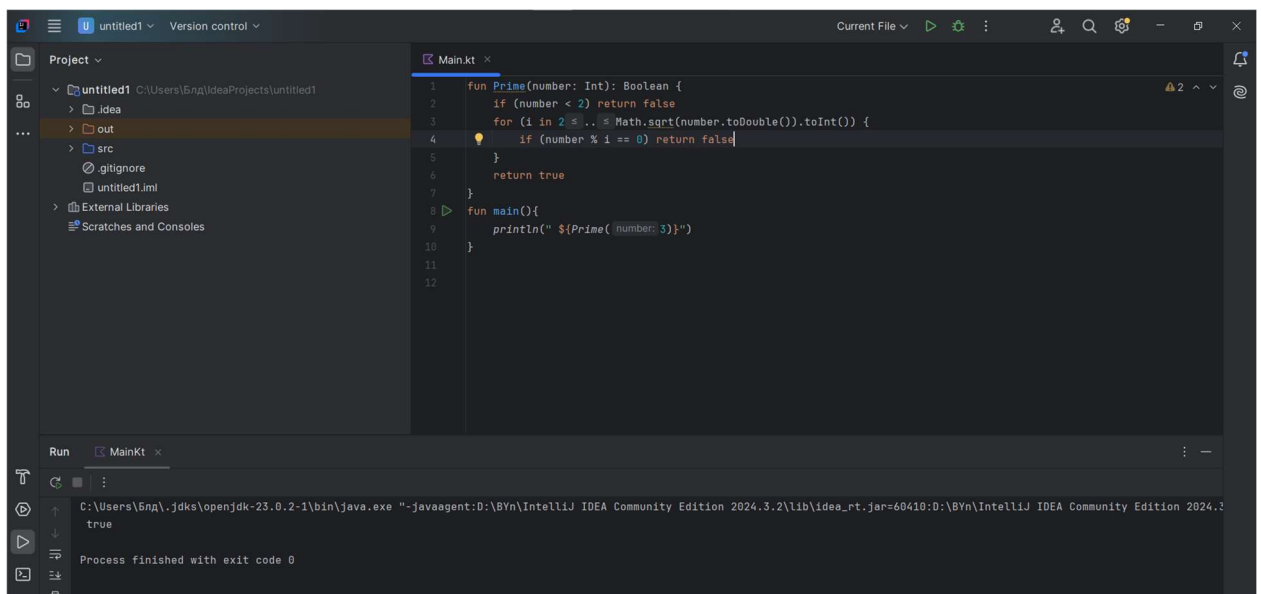
```
fun i(number: Int): Boolean {
    return number % 2 == 0
}
```

17. Факториал числа: Реализуйте функцию, которая вычисляет факториал введенного числа.

```
fun fact(n: Int): Int {
    return if (n <= 1) 1 else n * fact(n - 1)
}
```



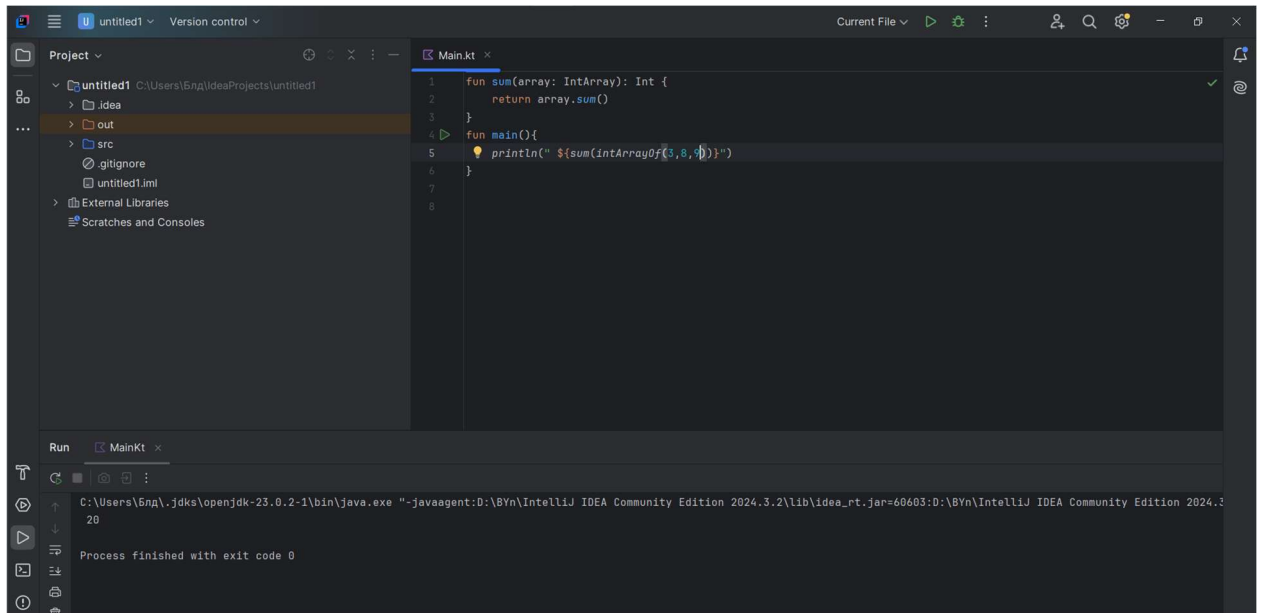
18. Проверка на простоту: Напишите функцию, которая проверяет, является ли заданное число простым.



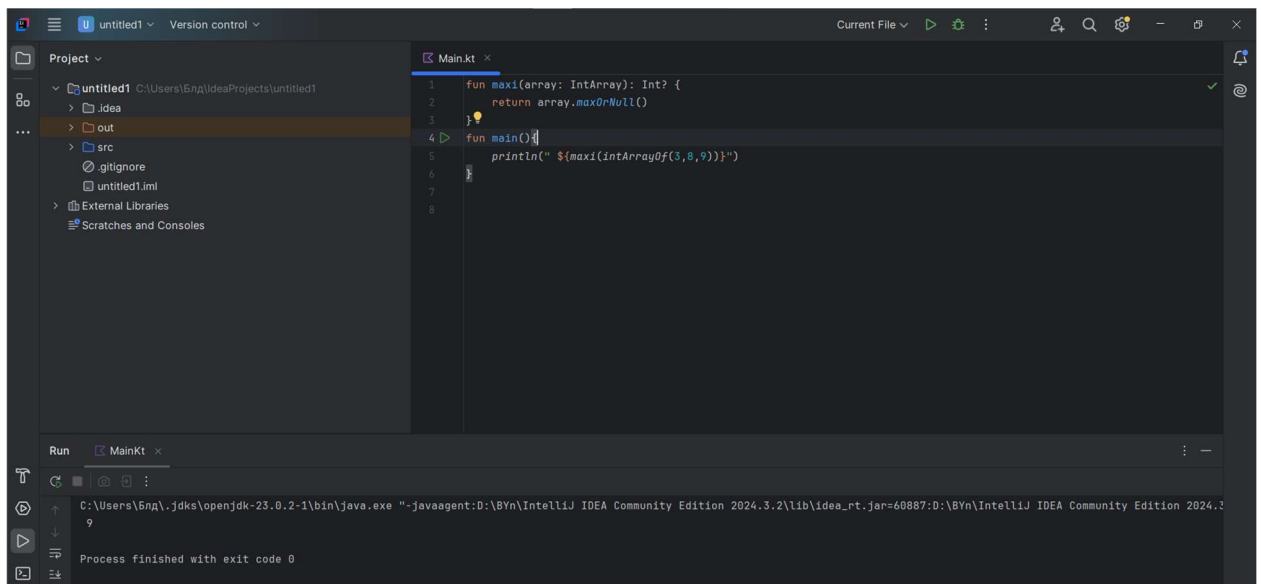
```
fun Prime(number: Int): Boolean {  
    if (number < 2) return false  
    for (i in 2..Math.sqrt(number.toDouble()).toInt()) {  
        if (number % i == 0) return false  
    }  
    return true  
}
```

19. Сумма чисел в массиве: Создайте функцию, которая принимает массив целых чисел и возвращает их сумму.

```
fun sum(array: IntArray): Int {  
    return array.sum()  
}
```

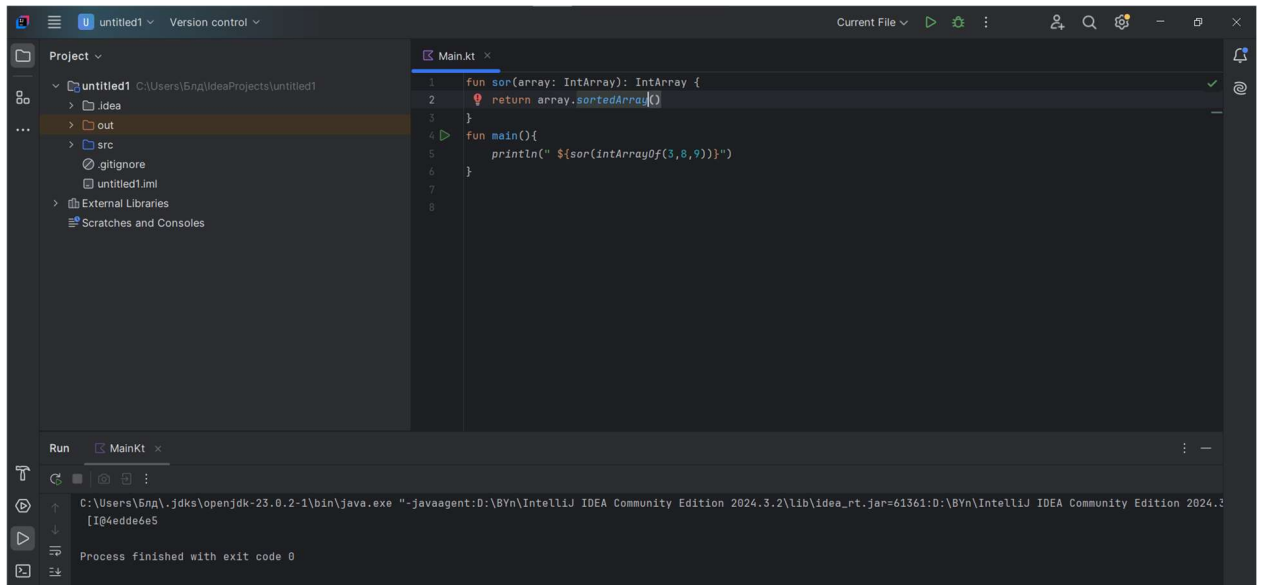
20. .Наибольшее число в массиве: Напишите функцию, которая находит и возвращает максимальное значение в массиве



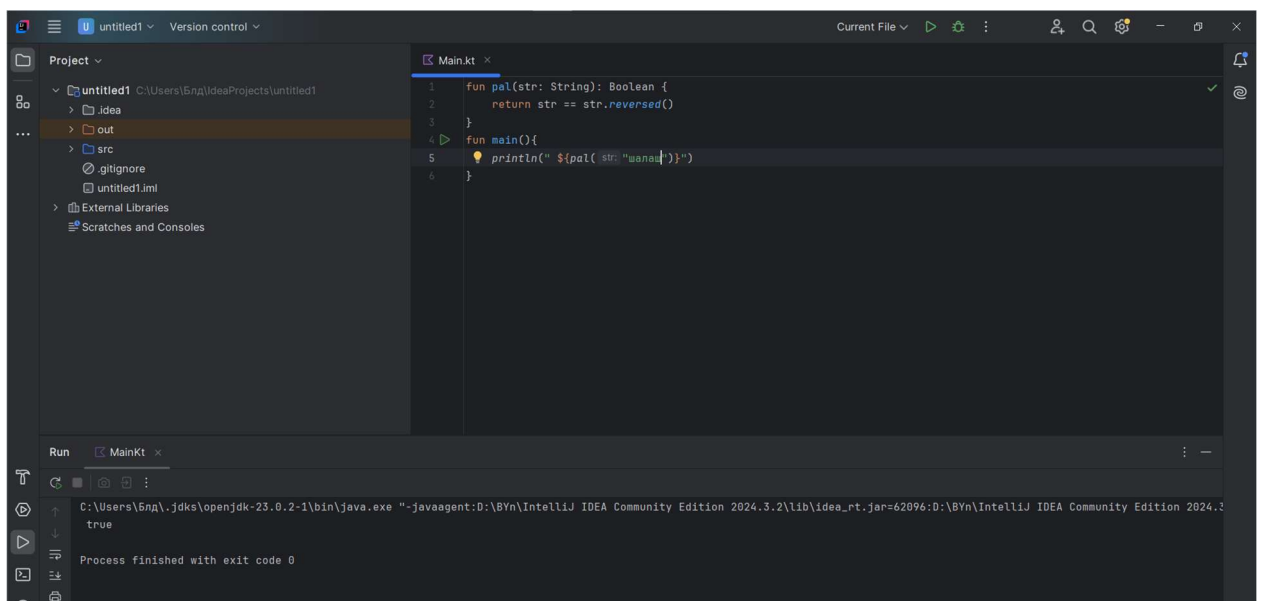
```
fun maxi(array: IntArray): Int? {  
    return array.maxOrNull()  
}
```

21. .Сортировка массива: Создайте функцию, которая сортирует массив чисел в порядке возрастания.

```
fun sor(array: IntArray): IntArray {  
    return array.sortedArray()  
}
```



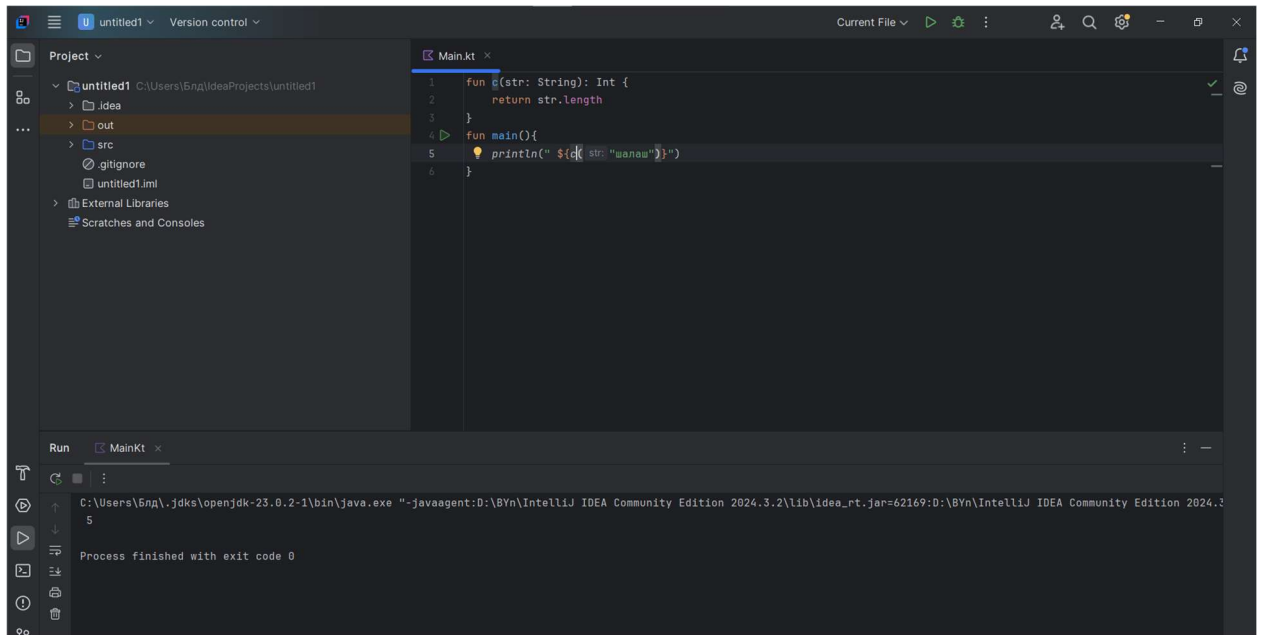
22. Проверка палиндрома: Напишите функцию, которая проверяет, является ли строка палиндромом.



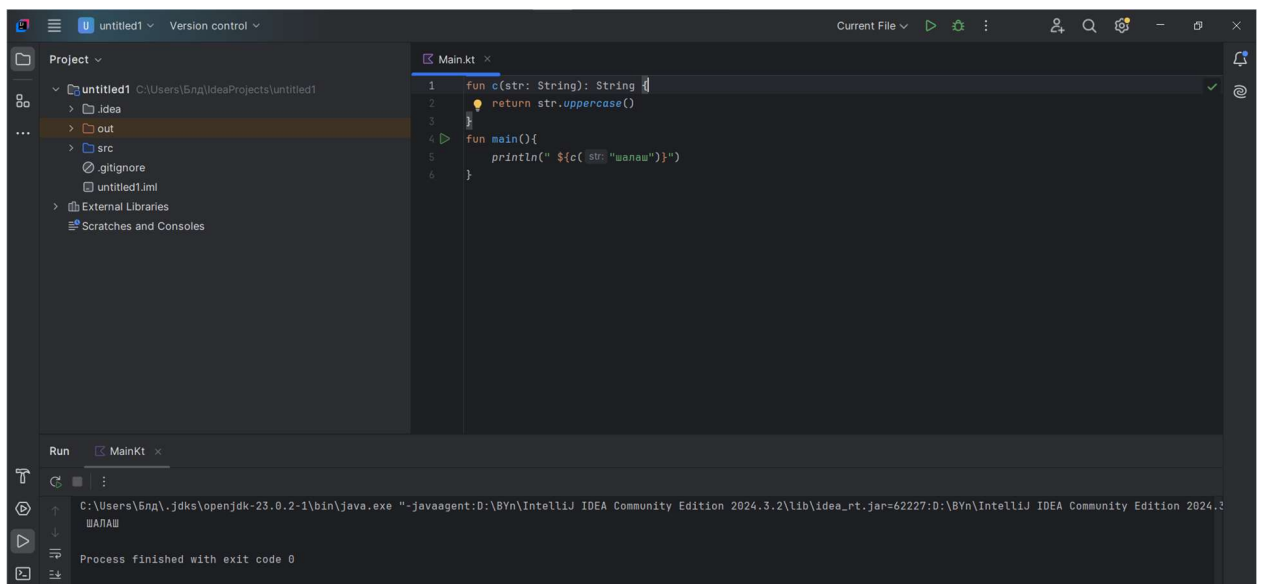
```
fun pal(str: String): Boolean {
    return str == str.reversed()
}
```

23. .Количество символов: Создайте функцию, которая принимает строку и возвращает количество символов в ней.

```
fun countCharacters(str: String): Int {
    return str.length
}
```



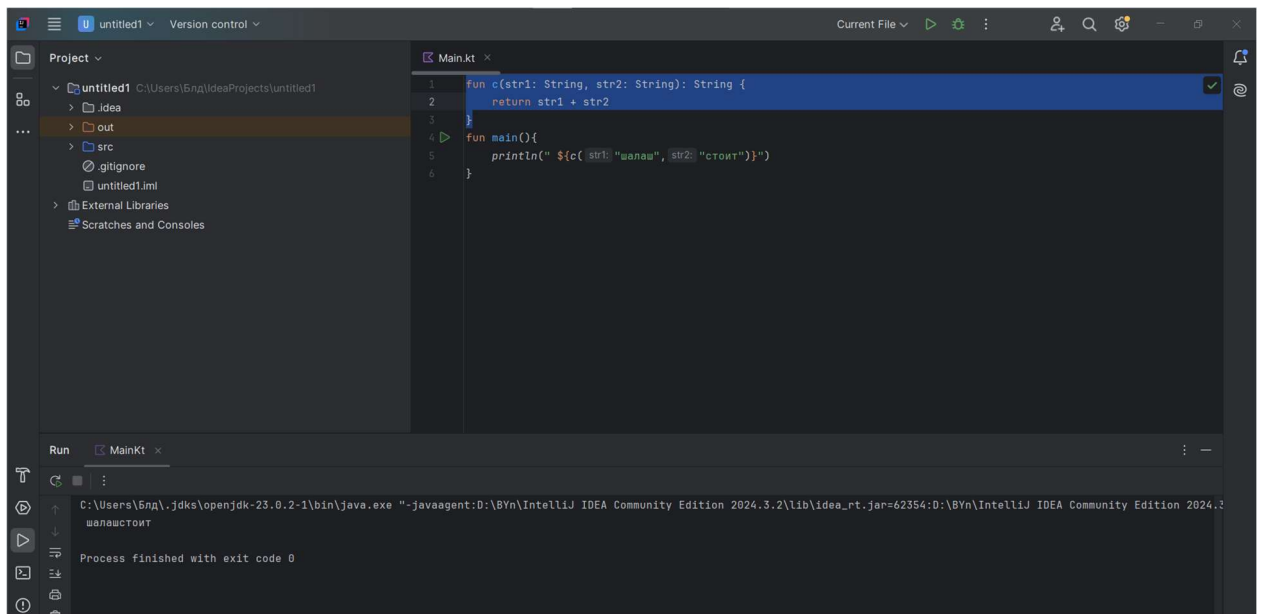
24. Конвертация в верхний регистр: Напишите функцию, которая принимает строку и возвращает её в верхнем регистре



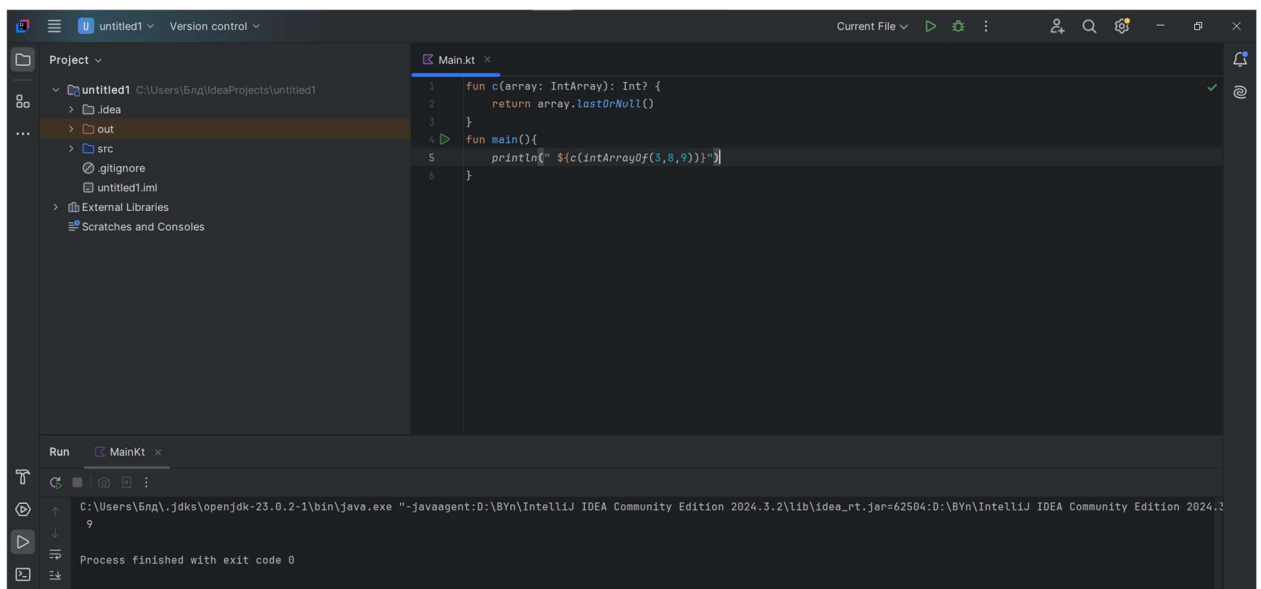
```
fun c(str: String): String {  
    return str.uppercase()  
}
```

25. Объединение строк: Создайте функцию, которая принимает две строки и возвращает их объединение.

```
fun c(str1: String, str2: String): String {  
    return str1 + str2  
}
```



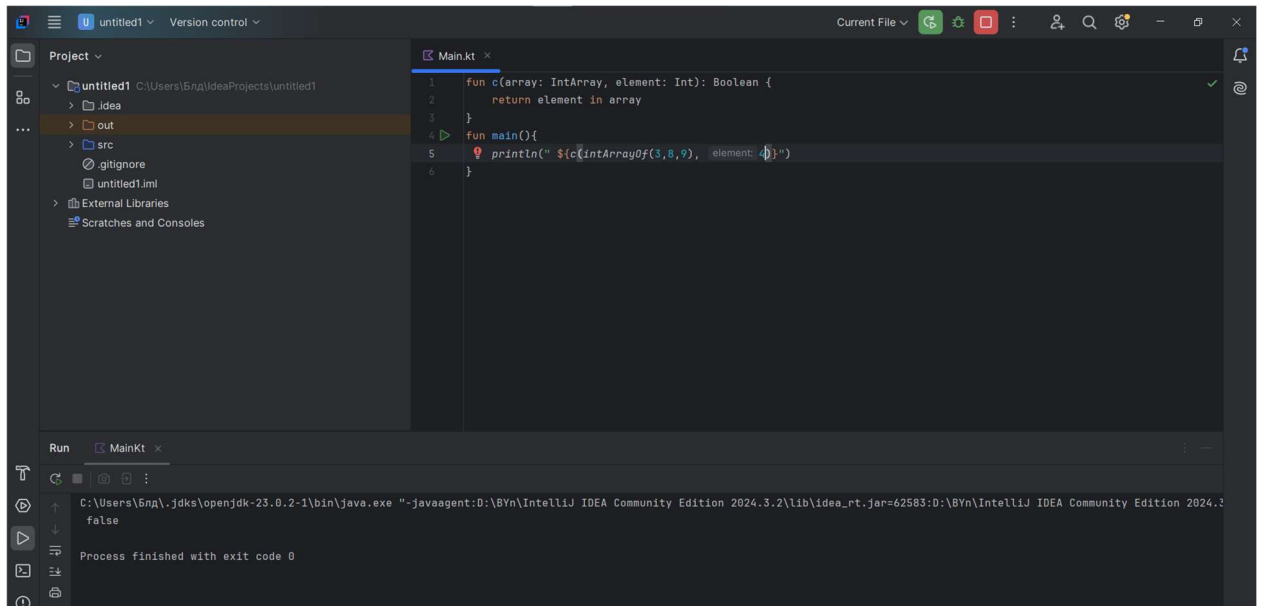
26. Возвращение последнего элемента массива: Напишите функцию, которая возвращает последний элемент переданного массива.



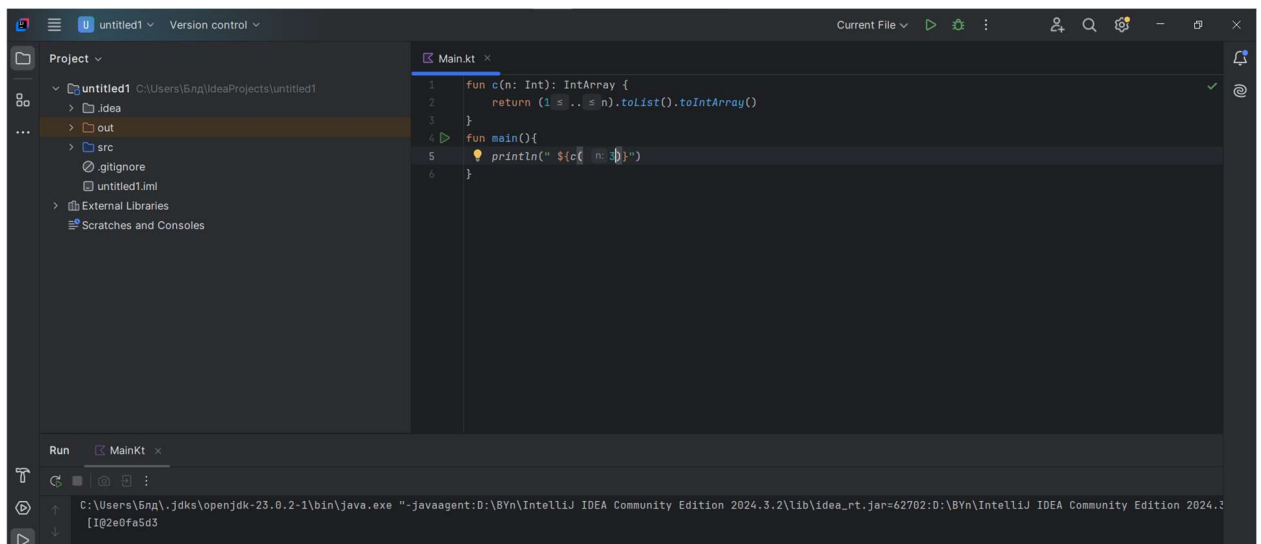
```
fun c(array: IntArray): Int? {  
    return array.lastOrNull()  
}
```

27. Проверка наличия элемента: Создайте функцию, которая проверяет, присутствует ли элемент в массиве

```
fun c(array: IntArray, element: Int): Boolean {  
    return element in array  
}
```



28. Создание массива от 1 до N: Напишите функцию, которая создает массив целых чисел от 1 до N



```

fun c(n: Int): IntArray {
    return (1..n).toList().toIntArray()
}

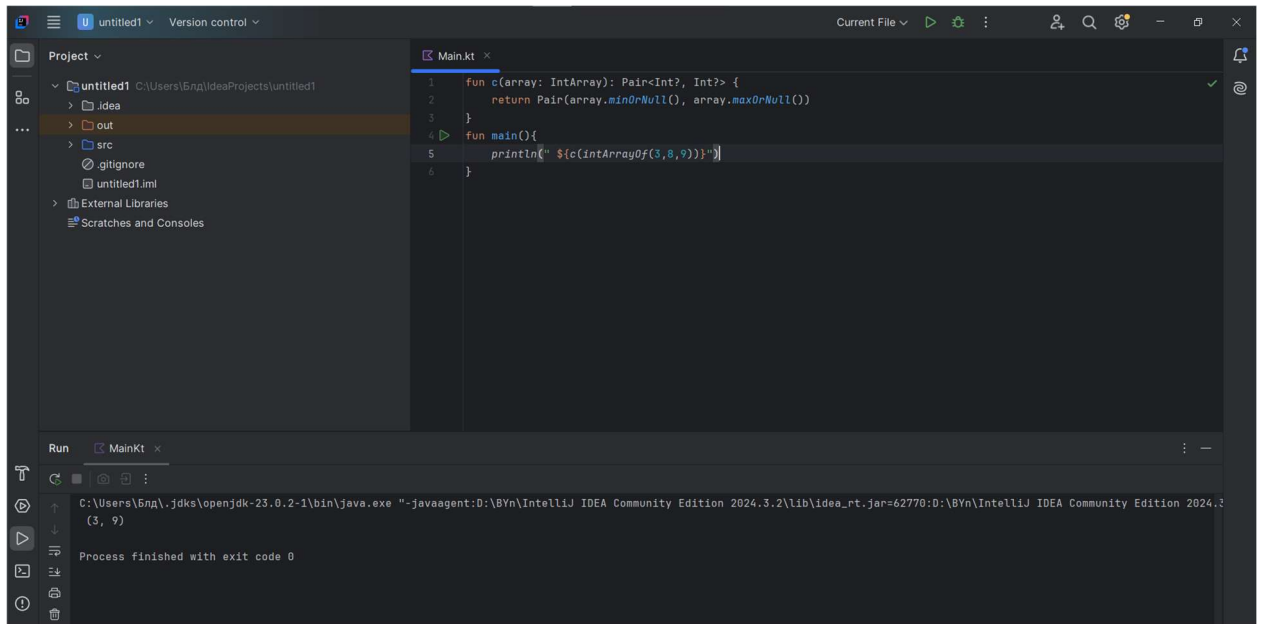
```

29. Максимум и минимум: Создайте функцию, которая находит одновременно максимальное и минимальное значения в массиве.

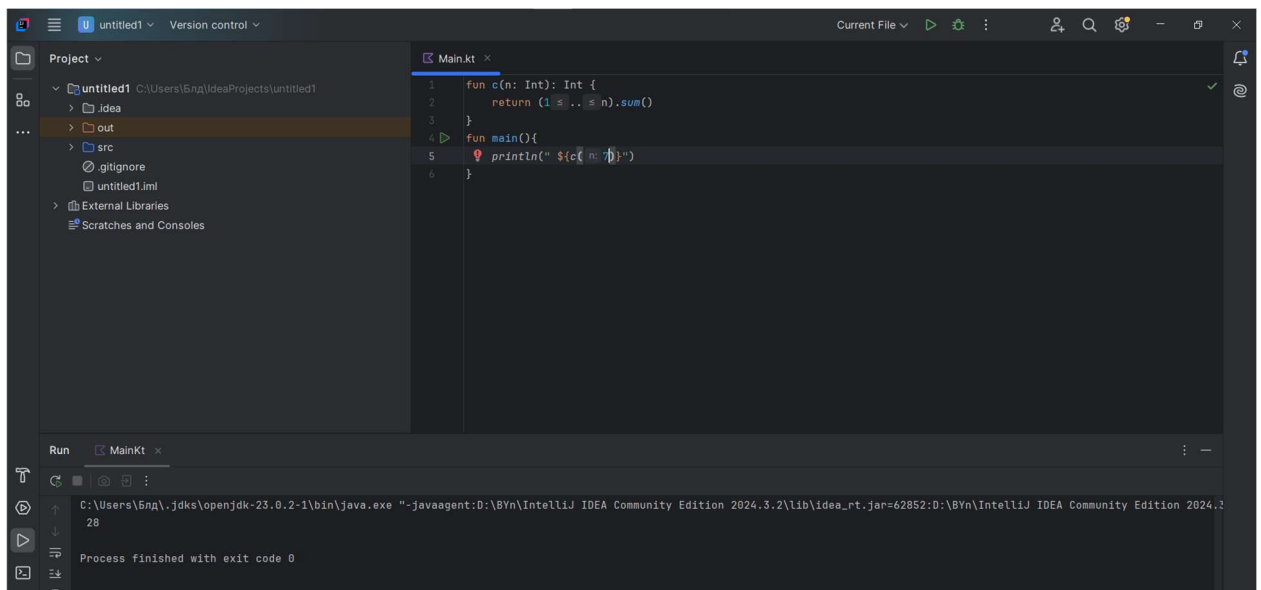
```

fun c(array: IntArray): Pair<Int?, Int?> {
    return Pair(array.minOrNull(), array.maxOrNull())
}

```



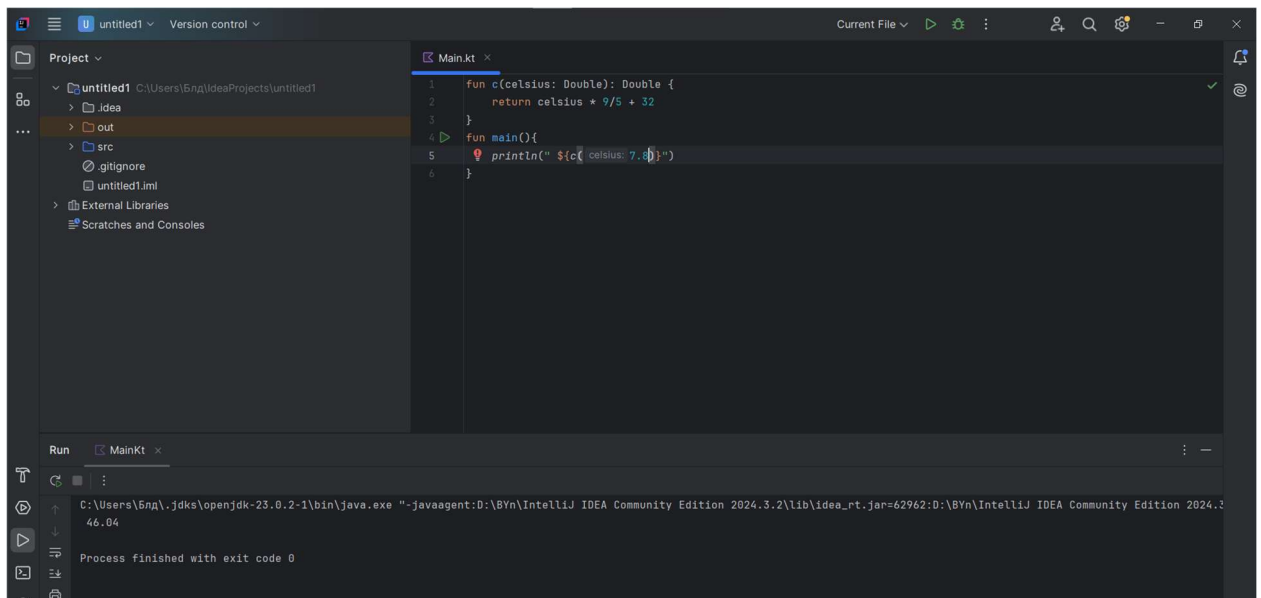
30. Сумма чисел от 1 до N: Напишите функцию, которая рассчитывает сумму всех чисел от 1 до N.



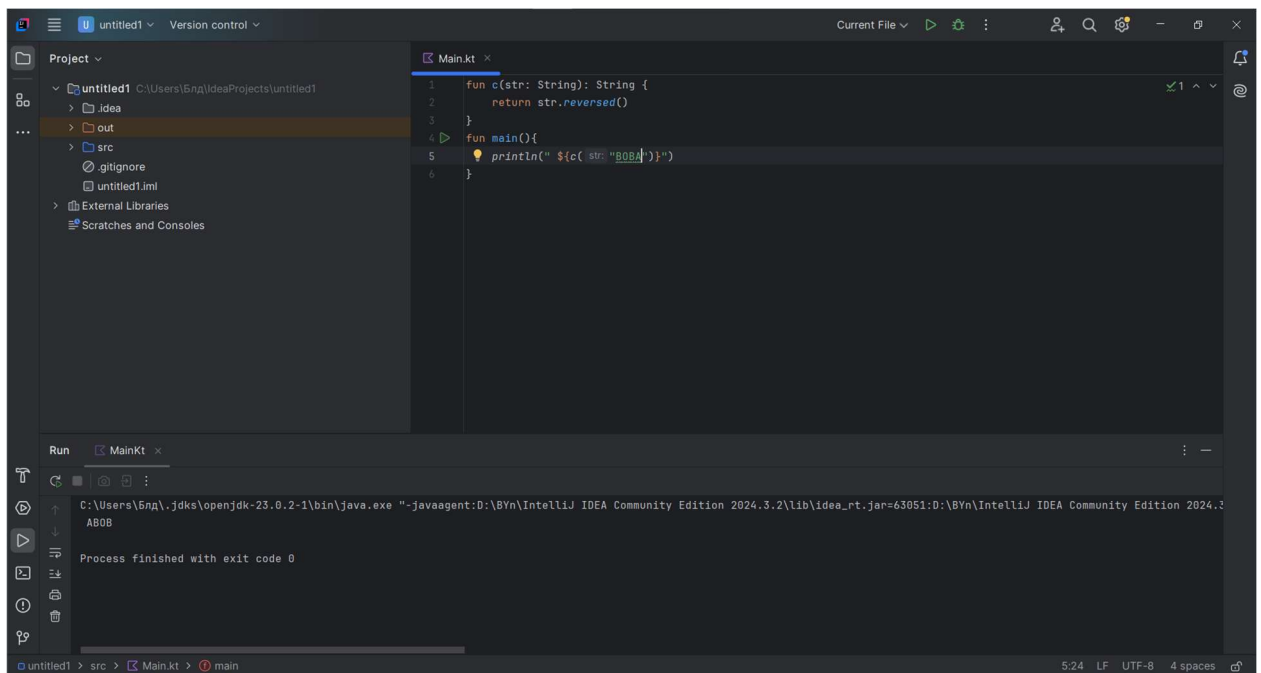
```
fun c(n: Int): Int {  
    return (1..n).sum()  
}
```

31. Преобразование Celsius в Fahrenheit: Создайте функцию, которая конвертирует температуру из Цельсия в Фаренгейт.

```
fun c(celsius: Double): Double {  
    return celsius * 9/5 + 32  
}
```



32. Обратный порядок строки: Напишите функцию, которая принимает строку и возвращает её в обратном порядке.



```

fun c(str: String): String {
    return str.reversed()
}

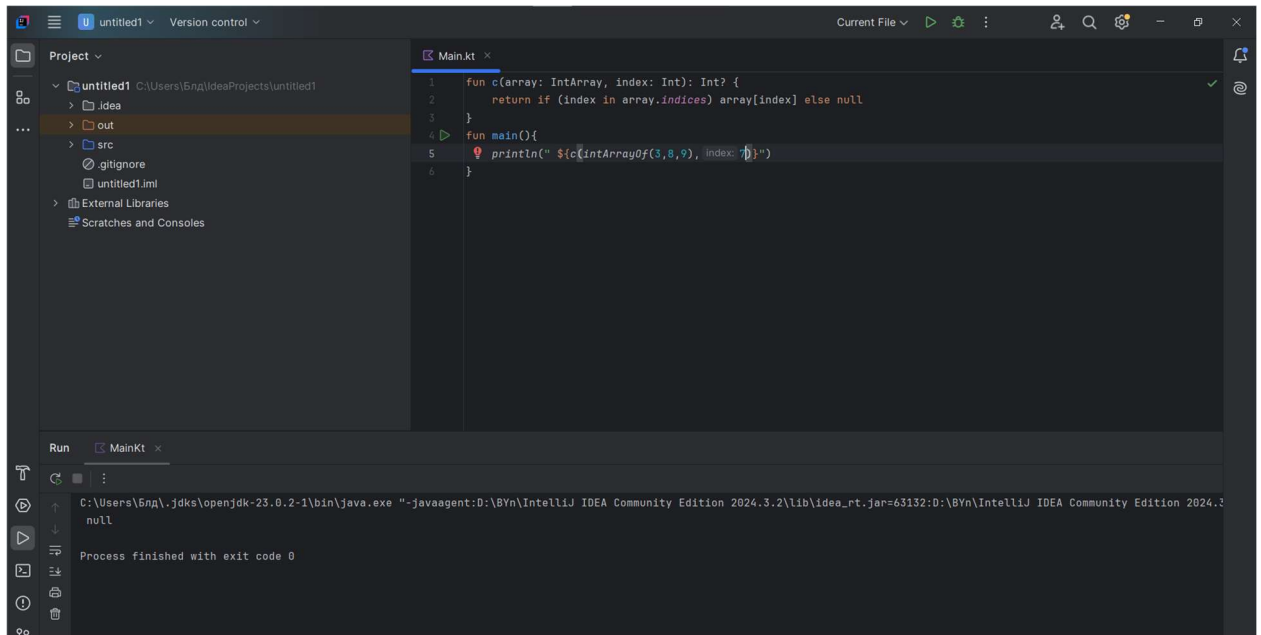
```

33. Поиск элемента по индексу: Создайте функцию, которая находит элемент по индексу в массиве.

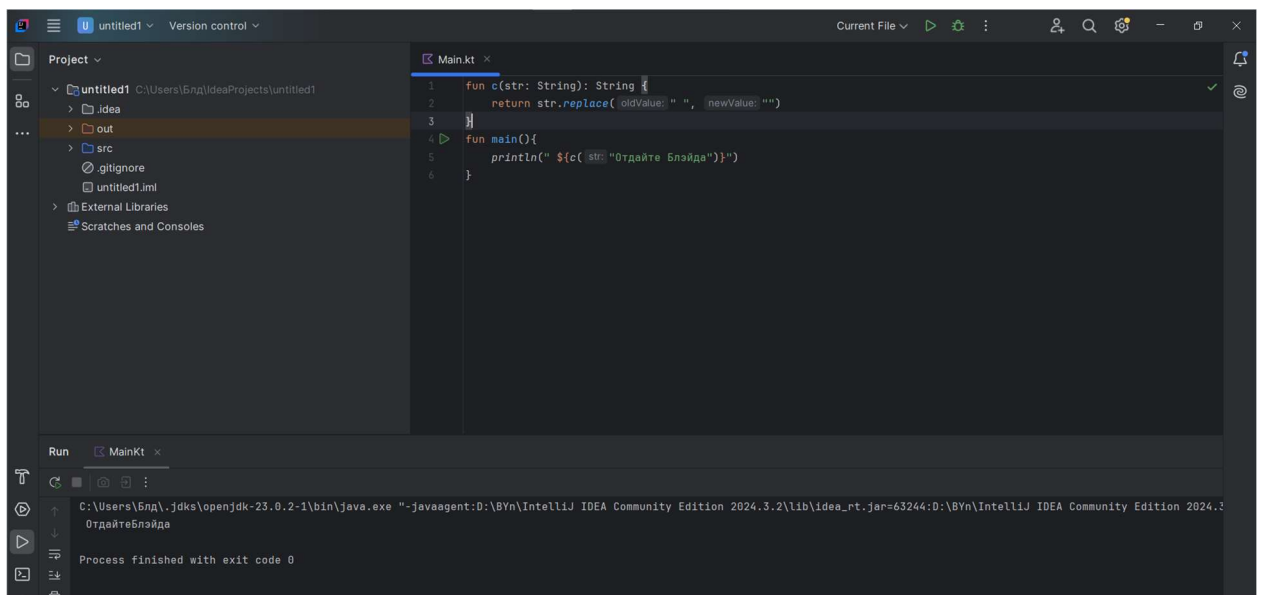
```

fun c(array: IntArray, index: Int): Int? {
    return if (index in array.indices) array[index] else null
}

```



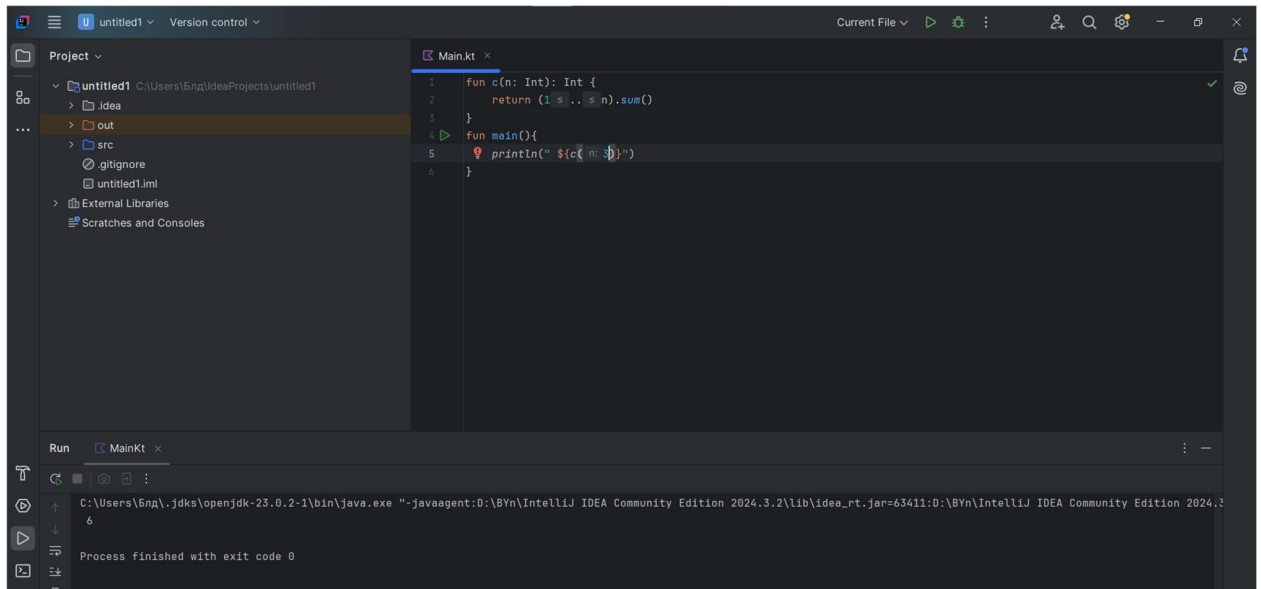
34. Удаление пробелов из строки: Напишите функцию, которая удаляет все пробелы из переданной строки.



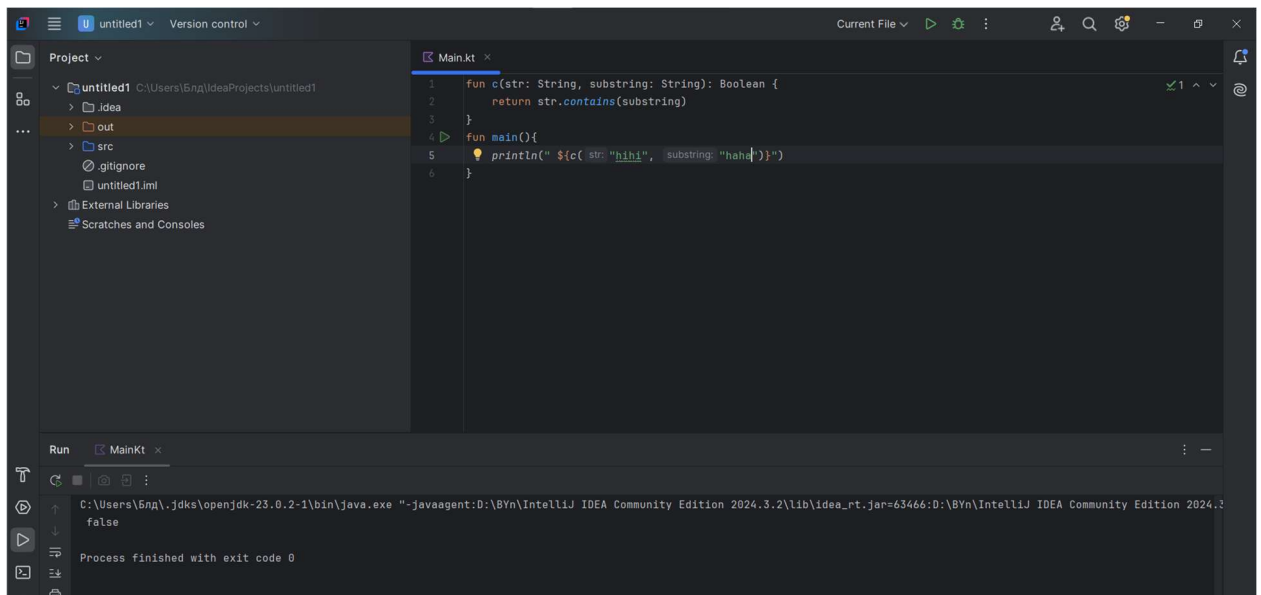
```
fun c(str: String): String {
    return str.replace(" ", "")
}
```

35. Сумма первых N натуральных чисел: Создайте функцию, которая возвращает сумму первых N натуральных чисел.

```
fun c(n: Int): Int {
    return (1..n).sum()
}
```

36. Проверка строки на наличие подстроки: Напишите функцию, которая проверяет, содержится ли одна строка в другой.



```

fun c(str: String, substring: String): Boolean {
    return str.contains(substring)
}

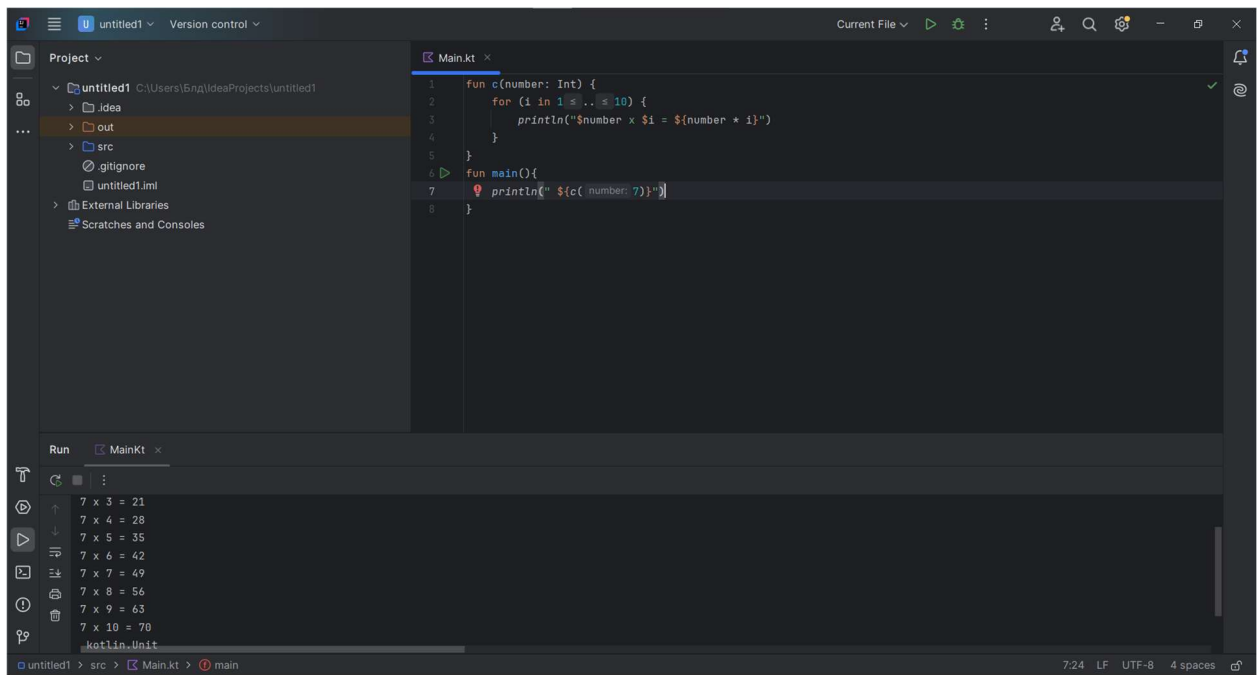
```

37. Печать таблицы умножения: Создайте функцию, которая выводит таблицу умножения от 1 до 10 для заданного числа

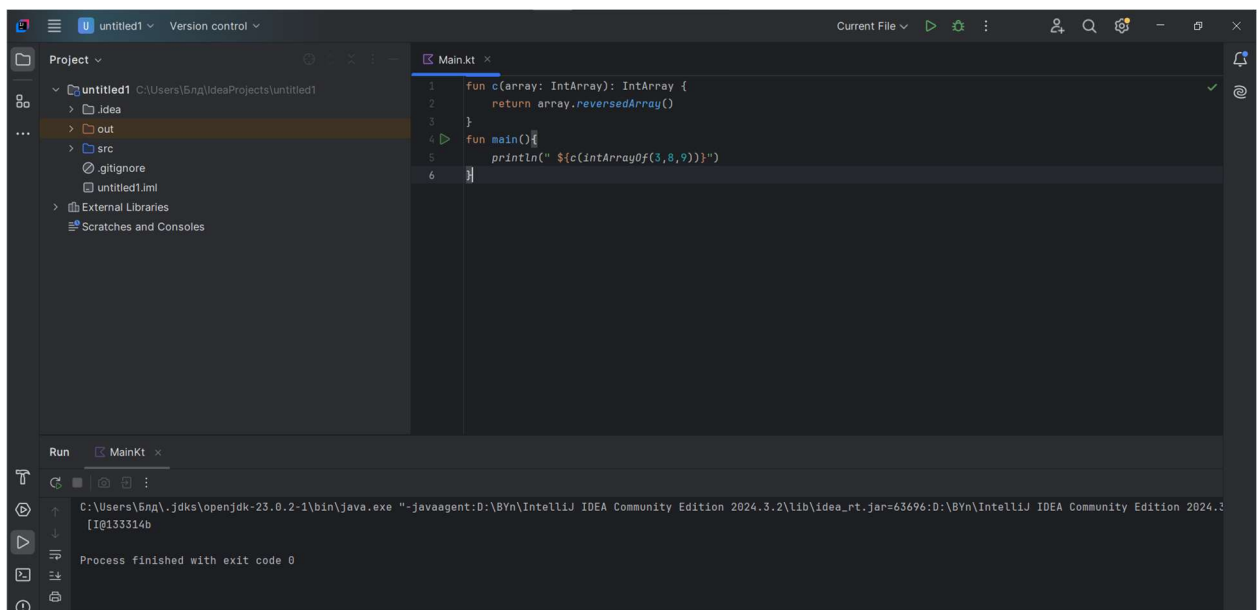
```

fun c(number: Int) {
    for (i in 1..10) {
        println("$number x $i = ${number * i}")
    }
}

```



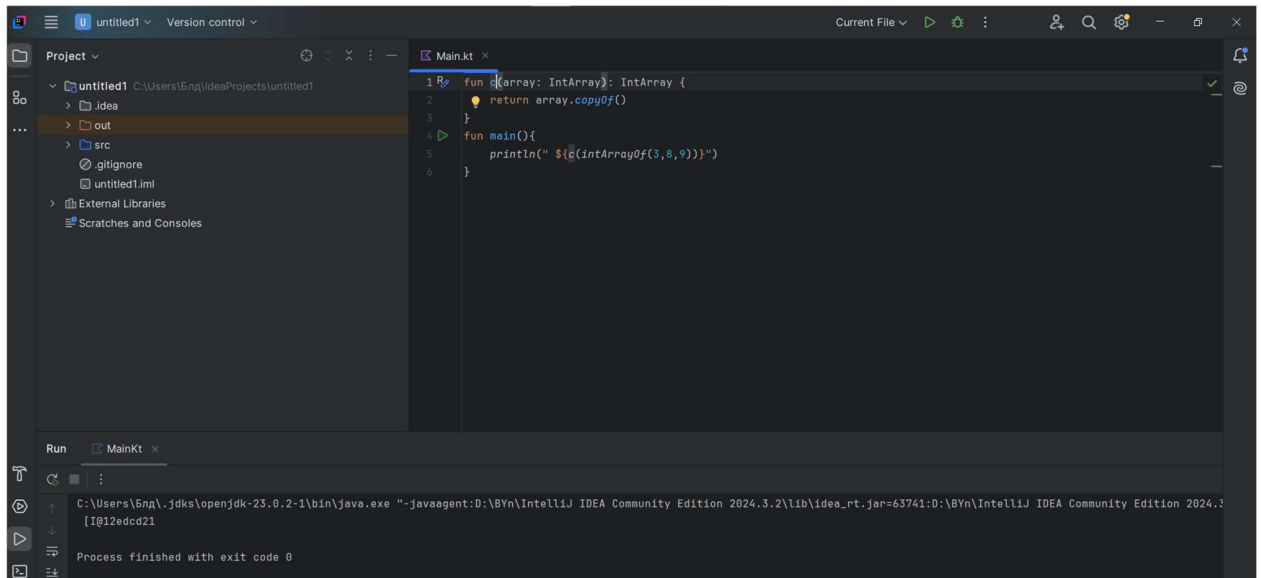
39. Переворот массива: Создайте функцию, которая переворачивает массив целых чисел.



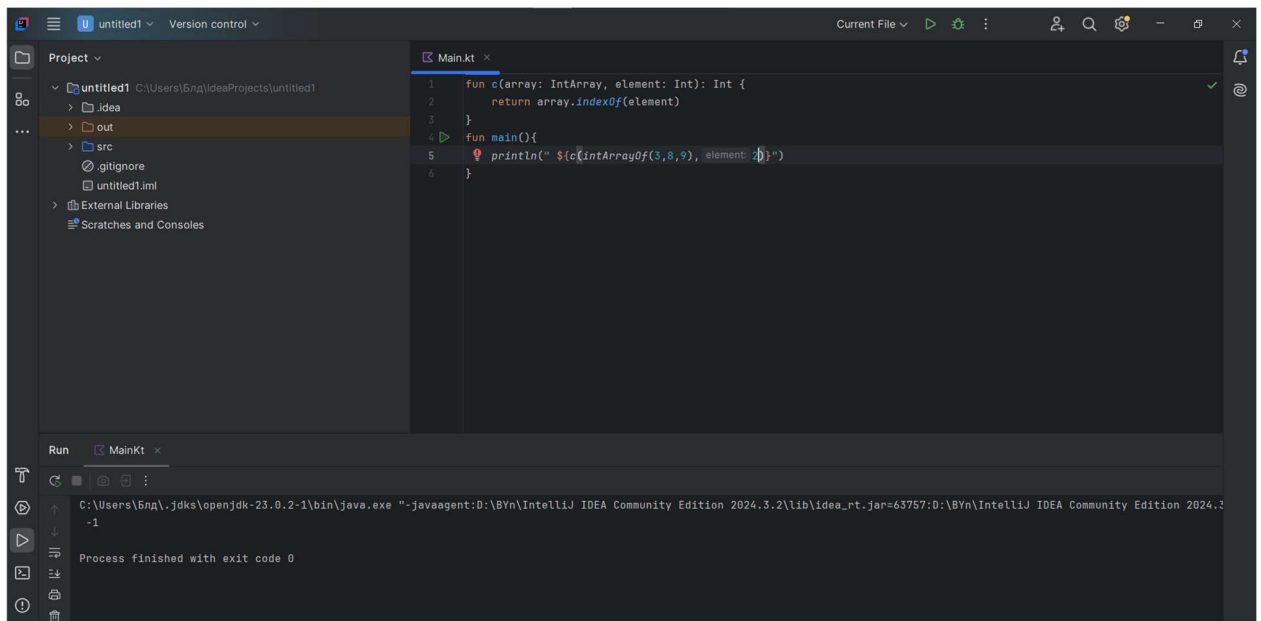
```
fun c(array: IntArray): IntArray {  
    return array.reversedArray()  
}
```

40. Копирование массива: Напишите функцию, которая копирует массив и возвращает новый массив.

```
fun c(array: IntArray): IntArray {  
    return array.copyOf()  
}
```



42. Индекс первого вхождения: Напишите функцию, которая возвращает индекс первого вхождения элемента в массиве, или -1, если элемент не найден.



```
fun c(array: IntArray, element: Int): Int {  
    return array.indexOf(element)  
}
```

41. Количество гласных в строке: Создайте функцию, которая считает количество гласных в строке.

```
fun c(str: String): Int {  
    val vowels = "aeiouAEIOU"  
    return str.count { it in vowels }  
}
```

